Steepest Descent vs. Newton Direction

Dong Nie

Formulation: $\min f(x)$

1) Directions:

An initial thought: Steepest descent: $p_k = -\nabla f_k$

The negative direction of gradient. What's the gradient direction? Just consider two dimension: It's the normal direction at point (x,y). (We can take a circle as an example, its gradient is (2x,2y), so its direction is normal direction (orthogonal to the tangent direction). When it comes to one dimension, the gradient's direction is positive/negative.

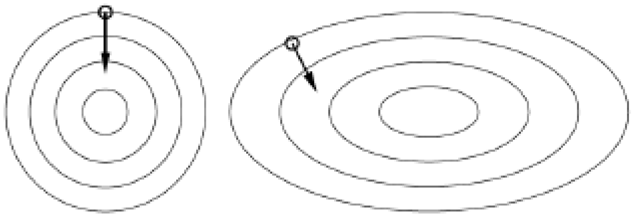

Fig. 1 Example of steepest descent

It's very simple, while this direction is often far from optimal for convergence to the minimum, as shown in Fig. 1.

Any other direction p_k that can decrease f(x). One important case is the Newton direction. The Newton direction is obtained by a locally quadratic approximation of the problem (2-nd order Taylor expansion).

$$f(x_k + p) \approx f_k + p^T \nabla f_k + \frac{1}{2} p^T \nabla^2 f_k p$$

where $\nabla^2 f$ denotes the Hessian of f. So we suppose p is the direction which minimizes the quadratic function. (We suppose $f(x_k + p)$ should be smaller than f_k). How to choose p will give us the smallest value of $f(x_k + p)$ (which means, decrease the most from f_k). We only have to take differentiable for p, then we get:

$$\frac{d}{dp} f(x_k + p) \approx \nabla f_k + \nabla^2 f_k p = 0,$$

From which follows

$$p_k^N = -(\nabla^2 f_k)^{-1} \nabla f_k$$

This is the Newton direction. This direction can be used as the search directions for the line search, $\nabla^2 f_k$ is positive definite (ie., has all positive eigenvalues). If it is not positive definite, the Hessian can be modified by directly or indirectly modifying its eigenvalues. E.g., by adding γI , where $r = \max(0, \delta - \omega_{min}(\nabla^2 f_k))$

Note, though Newton Direction converges at O(N^2), it is much faster than Steepest Descent (O(N)). While Newton Driection is not so widely used due to some limitations: 1). Require 2nd order differentiable 2). Require matrix invertible 3). If the matrix is too large, we cannot easily handle the inverse of matrix (e.g., an image is m*n pixels, then Hessian matrix is (mn)^2 elements)

2) How far do we go? (step size)

This problem will be explored in first order primal-dual algorithms.

To be continued…